

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

REMARKS/ARGUMENTS

Claims 1-21 were pending and were variously rejected under 35 USC §102(e) as being anticipated by Barnett in view of Official Notice. Claim 19 was canceled, thus claims 1-18, 20 and 21 are now pending. In light of the remarks below, the undersigned respectfully traverses the rejections.

I. INITIAL MATTERS

Claims 1-21 were also provisionally rejected under the judicially created doctrine of obviousness-type double patenting over claims 1-21 of copending Application No. 09/834,851.

In response, the undersigned respectfully requests that this provisional rejection be held in abeyance. If either or both of the copending applications are issued as patents before the present application issues as a patent, the undersigned is prepared to provide a terminal disclaimer in response to a non-provisional double patenting rejection.

Various amendments were also made to the claims to more clearly recite Markush-type claims. Such amendments were not made for purposes of patentability.

II. THE PRESENT INVENTION

The present invention relates to methods and systems for specifying promotions and distributing promotions across a computer network relying upon a unique and novel software architecture and mechanisms.

Initially, the specification distinguishes "promotions" or "electronic incentives" used herein from conventional "coupons." As described in the specification, page 15, lines 3-7:

These promotions are not considered "coupons" as "coupons" is understood in the industry. More specifically, in the industry, "coupons" are typically defined as detachable certificates, tickets, or the like that entitle the bearer or holder to a benefit. In the present embodiment, the customer and the merchant server are not given any such detachable and/or possessable certificate and cannot hold, bear, or present anything.

Additionally, the specification notes that coupons require possession of a cookie or the like:

By way of contrast, in one electronic couponing systems, a electronic coupon describing a right or benefit is created in a couponing server. The electronic coupon, or token, is then downloaded to a customer's computer system and stored. These coupons or tokens may be in the form of a cookie or the like stored on the customer's computer system. Much later, the customer may enter an electronic store that is independent of the electronic couponing system. Next, the cookie or token stored on the customer's computer system is retrieved and passed back to the electronic store web server. Because the customer's computer had "possession" of the cookie or token in the computer memory, the electronic store web server provides the customer the right or benefit or the bargain described, i.e. the customer is entitled to a 10% discount. This example thus illustrates that the electronic cookie or token incorporates the standard "coupon" model: the

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

customer's computer memory stored the cookie, and possession of the cookie was a condition for receiving the bargain.

The background of the invention describes some problems with these possessable coupons. More specifically, one problem is that coupons for a product may be provided to users who were already going to buy the product, p2, lines 1-12:

A problem with traditional coupons includes that coupons often end up in the hands of buyers who are not targeted. This is because distributing coupons only to target buyers is virtually impossible. Although some coupons may be distributed to channels such as magazines, direct mailings, and the like that include a large percentage of target buyers, a significant percentage nevertheless reaches non-target buyers. These non-target buyers may include those willing to purchase the product even without the coupon. Accordingly, if non-target buyers uses the coupons to purchase a product, this directly reduces the amount of profit to the promoter. As an example, a promoter may create a promotion directed to Pepsi™ drinkers to try Coke™. To do so, the promoter offers coupons providing the bearer with a dollar off a six-pack of Coke™. However, it is virtually impossible to prevent a devoted Coke™ drinker from picking and redeem that coupon. This sort of common situation directly "siphons-off" manufacturer profits.

In light of this problem, the specification states that improved methods for providing targeted promotions are needed, without the problems highlighted above.

Many of the amendments to the claims and distinctions over the cited art depend upon an understanding of the following specific software concepts: As expressly described in the specification, "object-oriented" software programming techniques are used, p. 13, lines 27-28, such as Microsoft COM software objects. For example, service objects, coupon objects, product objects, are described and used.

The specification should be read and claims should be interpreted in light of the object-oriented environment described. Particular terms related to object-oriented software were defined and / or used in the specification consistently with how these terms are used in the software industry. The definitions of such terms in the software industry may override non-technical dictionary definitions of such words. For the Examiner's reference, particular definitions of terms are reproduced from the Microsoft Press Computer Dictionary, second edition, 1994 in attachment A to this amendment: object-oriented programming, object, instance, instantiate, and class.

Discussion of specific embodiments will be described below:

On p. 15, lines 32-35, the specification describes the merchant server invoking a Service object:

[T]he merchant server invokes a Service object within the application server to evaluate the customer's shopping category to determine if there [are] any coupons to display, step 560.

On p. 15, line 35 - p. 16, line 2, the specification describes the application server instantiating coupon objects:

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

In response to the current shopping category, the application server determines whether any promotions are applicable and if so, one or more "Coupon Objects" are instantiated, step 570.

On p. 16, lines 13-17, the specification describes the merchant server querying the instances of the coupon objects:

Next, merchant server 140 queries one or more "Coupon Objects" that have been instantiated for a description of the pre-conditions and benefit, a[n] image of the product, and the like, step 620. In response, merchant sever 140 specifies the rendering of the promotion on an HTML page for display on the customer's display, step 630.

On p. 17, lines 10-14, the specification describes the merchant server invoking another service object;

When the consumer desires to checkout, merchant server 140 causes application server 180 to use the instances of "Coupon Objects" that were created, step 710. In particular, an evaluate method of a Service object is invoked, and the amount of savings is calculated. The savings is then retrieved by merchant server 140 and displayed to the consumer, step 715.

In the present embodiment, when the consumer checks out, a promotion usage condition, application server 180 stores data associated with the transaction, step 720. The savings [are] then retrieved by merchant server 140 and displayed to the consumer, step 715.

The claims, as amended, incorporate at least some of the object-oriented concepts discussed above. For example, claim 1, now recites:

receiving usage data of the electronic incentive from the application server, wherein the usage data is determined in response to a promotion usage condition of an instance of the electronic incentive indicated by the application server;

wherein the instance of the electronic incentive is instantiated in response to a invocation of a method on a service object stored in the application server by the merchant server;

wherein the instance of the electronic incentive is stored in the application server and is queried by the merchant server;

wherein the merchant server specifies rendering of the data associated with the electronic incentive in response to a query of the instance of the electronic incentive;

wherein the promotion usage condition is indicated in the application server when a user coupled to the merchant server fulfills pre-conditions of the instance of electronic incentive.

For example, claim 8, now recites:

invoking an evaluation service object within an application server coupled to the merchant server for promotions, wherein an instance of a promotion object is created in the application server in response thereto ;

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

querying the instance of the promotion object within the application server for a description of the promotion, wherein the description includes pre-conditions, a user benefit and an output representation of the promotion;

invoking a savings method in a service object within the application server to determine a savings amount, wherein the savings amount comprises the user benefit when the selection of the at least one item fulfills the pre-conditions.

For example, claim 15, now recites:

creating an instance of the electronic incentive for the user in response to an invocation of an evaluation service object by a merchant server for determining electronic incentives for a user;

receiving a query for a description of the instance of the electronic incentive from the merchant server wherein the description includes the pre-condition and the benefit;

receiving an invocation of an amount of savings method of a service object from the merchant server to determine a savings for the user, wherein when an item placed in a shopping cart associated with the user in the merchant server fulfills the pre-condition, the amount of savings method indicates the benefit to the user in the merchant server.

III. BARNETT

Barnett is described as a method and system for the electronic distribution of coupons to consumers. Specifically Barnett appears to describe methods and systems where coupons bundles are provided to consumers via service providers.

Importantly Barnett does not refer to using an object-oriented paradigm. Instead, Barnett appears to simply rely upon simple procedural calls.

Additionally, Barnett appears to only refer to providing and redeeming conventional "coupons." Barnett, Fig. 9 includes a sample flow chart. In one step, the remote computer receives and stores variable "coupon data." Next, the coupon data is printed out and redeemed in-person, or the coupon is electronically redeemed. More specifically, the specification states on col. 9, lines 41-45.:

The requested coupon data package and associated advertising materials are transmitted by the online service provider 2 to the personal computer 6, where it is stored in the downloaded coupon data file 30a in the coupon database.

Next, the user prints out the coupons for redemption, col. 10 lines 58-60:
Coupons are printed by the printable coupon data generation routine 32d, which is invoked by a user when he selects a print command from the coupon file function 56.

In the case of electronic redemption, the coupon is electronically transferred, col. 1, lines 38-44.:

This is especially useful in the "electronic shopping mall" environment now found in many online services. The electronic coupon data could also

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

be routed via the data communications interface 20 to a retail store where the user will be shopping, where the coupon data is held in a buffer pending purchase by the user of the matching product.

To address the problem of unauthorized use or duplication of these coupons, Barnett describes using user-specific data in a bar code 90. Col. 7, line 24-25. Further, Barnett describes:

The unique user bar code 90 also renders the electronic coupon system of the present invention secure and virtually fraud-proof. Although a user is able to print out a particular coupon 18 only once (to be described in detail below), the coupon issuer 14 could still be defrauded by a user or retailer who might photocopy a printed coupon numerous times and fraudulently and repeatedly present it for redemption. However, in accordance with the present invention, each coupon printed by a user is unique, and the scanning of a coupon presented for redemption will be stored at the coupon redemption center. Thus, the coupon issuer will know if a particular user has redeemed a particular coupon and thus disallow further redemption of a photocopied coupon bearing the same indicia.

However, in Barnett, because a user possesses these coupons, a user may still print-out a coupon for a product and give it to another user, who would have purchased the product even without the coupon. Accordingly, the manufacturer's profits may still undesirably be "siphoned-off" by these actions.

IV. BARNETT DISTINGUISHED

A. Claim 1

Barnett fails to disclose every element of claim 1.

More particularly, Barnett fails to disclose receiving usage data of the electronic incentive from the application server, wherein the usage data is determined in response to a promotion usage condition of an instance of the electronic incentive indicated by the application server.

Further, Barnett fails to disclose wherein the instance of the electronic incentive is instantiated in response to a invocation of a method on a service object stored in the application server by the merchant server, and wherein the instance of the electronic incentive is stored in the application server and is queried by the merchant server.

Additionally, Barnett fails to disclose wherein the merchant server specifies rendering of the data associated with the electronic incentive in response to a query of the instance of the electronic incentive, and wherein the promotion usage condition is indicated in the application server when a user coupled to the merchant server fulfills pre-conditions of the instance of electronic incentive.

As discussed above, Barnett fails to disclose anything about an implementation using an object-oriented approach and / or objects. Barnett simply describes that coupon data are simply downloaded from a online service provider to a user at a personal computer. Once the

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

coupon data is on the personal computer in Barnett, the on line service provider loses control of the coupon.

In contrast, the claimed limitations describe the merchant server querying instances of coupon objects stored on an application server, not disclosed in Barnett. Further, as recited above, the application server determines whether the promotion usage condition is fulfilled or not.

In light of the above, and for other reasons, Barnett fails to disclose all elements of claim 1. Accordingly, Barnett does not anticipate claim 1.

B. Claim 8

Barnett fails to disclose every element of claim 8. More specifically, Barnett fails to disclose the limitations of invoking an evaluation service object within an application server coupled to the merchant server for promotions, wherein an instance of a promotion object is created in the application server in response thereto, querying the instance of the promotion object within the application server for a description of the promotion, wherein the description includes pre-conditions, a user benefit and an output representation of the promotion, and invoking a savings method in a service object within the application server to determine a savings amount, wherein the savings amount comprises the user benefit when the selection of the at least one item fulfills the pre-conditions.

As discussed above, Barnett fails to disclose anything about an implementation using an object-oriented approach and /or objects. Instead, Barnett simply describes that coupon data are simply downloaded from a online service provider to a user at a personal computer.

In contrast, the claim language above illustrates object-oriented concepts, not disclosed in Barnett, as well as specific usages of software objects.

In light of the above, and for other reasons, Barnett fails to disclose all elements of claim 8. Accordingly, Barnett does not anticipate claim 8.

C. Claim 15

Barnett fails to disclose every element of claim 15. More specifically Barnett fails to disclose a method for an application server including creating an instance of the electronic incentive for the user in response to an invocation of an evaluation service object by a merchant server for determining electronic incentives for a user, receiving a query for a description of the instance of the electronic incentive from the merchant server wherein the description includes the pre-condition and the benefit, and receiving an invocation of an amount of savings method of a service object from the merchant server to determine a savings for the user, wherein when an item placed in a shopping cart associated with the user in the merchant server fulfills the pre-condition, the amount of savings method indicates the benefit to the user in the merchant server.

As discussed above, Barnett fails to disclose anything about an implementation using an object-oriented approach and /or objects. Instead, Barnett simply describes that coupon data are simply downloaded from a online service provider to a user at a personal computer.

In contrast, the claim language above illustrates the object-oriented nature of embodiments of the present invention, not disclosed in Barnett, as well as the usage of such software objects within the application server.

Appl. No. 09/834,855
Amdt. dated August 12, 2005
Reply to Office Action of May 9, 2005

PATENT

In light of the above, and for other reasons, Barnett fails to disclose all elements of claim 15. Accordingly, Barnett does not anticipate claim 15.

D. Remaining claims

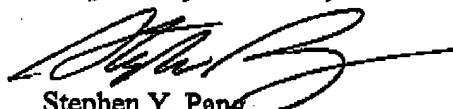
Claims 2-7; 9-14; 16-19 and 21, dependent upon claims 1, 8, and 15, respectively, are also asserted to be allowable for substantially the same reasons as claims 1, 8, and 15, respectively, and more specifically for the specific limitation they recite.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at (650) 326-2400.

Respectfully submitted,


Stephen Y. Pang
Reg. No. 38,575

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: (650) 326-2400
Fax: (650) 326-2422
Attachments: Appendix
SYP:deh

60499777 v1

USPTO
TO: Central Fax COMPANY:
AUG.12.2005 3:22PM

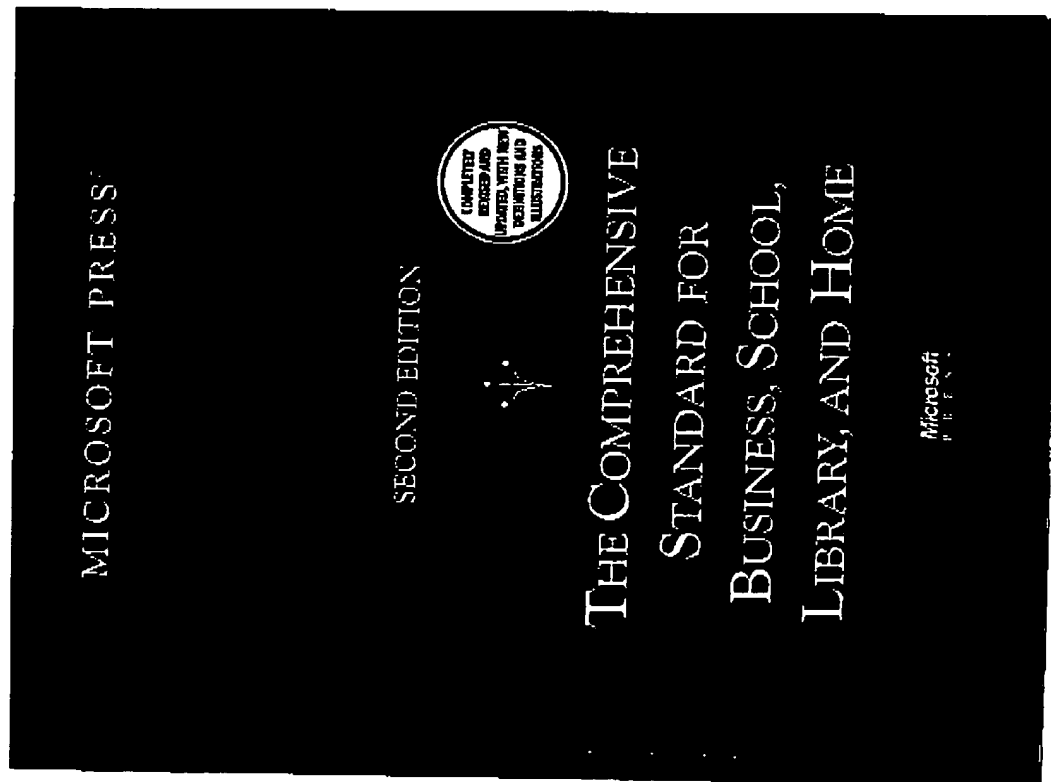
9/16/2005 10:59 AM PAGE 17/021
TTC-PA 650-326-2422

Fax Server
NO.095 P.16

APPENDIX

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98073-0999
Copyright © 1994 by Microsoft Press
All rights reserved. No part of the contents of this book may be reproduced or
transmitted in any form or by any means without the written permission of the publisher.
Library of Congress Cataloging-in-Publication Data
Microsoft Press computer dictionary : a computer reference standard for
business, school, library, and home / Microsoft Press. -- 2nd ed.
ISBN 1-55615-597-3
1. Computers--Dictionaries. 2. Microcomputers--Dictionaries.
I. Microsoft Press. II. This Computer Dictionary.
QP563.5.M64 1993
680.19--dc20
93-38669
CIP
Printed and bound in the United States of America
456789 0081 98765
Published in the book made in Canada by Macmillan of Canada, a division of Canada
Publishing Corporation.
Discovered in the book made outside the United States and Canada by
Penguin Books Ltd.
Penguin Books Ltd., Harmondsworth, Middlesex, England
Penguin Books Australia Ltd., Ringwood, Victoria, Australia
Penguin Books N.Z. Ltd., 242-250 Waiheke Road, Auckland 32, New Zealand
British Cataloguing in Publication Data available.
Project Editors: Casey D. Doyle
Manufacturing Editors: Alex C. Copp, Sarah
Technical Editors: Mary P. Long, Jeff Gurey, Don M. Jones, Jr., Jim Purkin, Seth Westbury

BEST AVAILABLE COPY



Object-oriented programming



Object-oriented programming (OOP). A programming paradigm in which a program is viewed as a collection of discrete objects that are self-contained collections of data structures and routines that interact with other objects. A class defines the data structures and routines of an object; an object is an instance of a class that can be used as a variable in a program. In some object-oriented languages, objects respond to messages, which are the principal means of communication. Other object-oriented languages use the traditional procedure-call mechanism. See also C++, object, Objective-C.

Object-oriented programming (OOP). A style of text control by which a program is not limited to the use of a single function but can be used to control a program's flow and to control the program's data.

OOP (Object-oriented programming). A style of text control by which a program is not limited to the use of a single function but can be used to control a program's flow and to control the program's data.

Binary	Decimal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Thus the binary number 01010001 can be converted to decimal by adding the values of the bits, as 001 010 011. Converted to decimal, the number becomes 123. Although, as this example shows, decimal numbers can be converted to binary, their values differ because of the different meanings assigned to each number position. In decimal notation, for

277

text or language that supports the use of objects. See also object.

Object-oriented graphics (OOG). Also called structured graphics. Computer graphics that are based on the use of "primitive elements" (graphics primitives), such as lines, curves, circles, and squares. Object-oriented graphics, used in applications such as computer-aided design and drawing, and illustration programs, describe an image mathematically as a set of instructions for creating the objects in the image. This approach contrasts with bitmapped graphics, the other widely used approach to creating images, which represents a graphic as a group of black and white or colored dots arranged in a matrix system. Object-oriented graphics enable the user to manipulate objects as entire units—for example, to change the length of a line or change a circle's radius. Whereas bitmapped graphics require replotting individual dots in the line or circle. Because objects are described mathematically, object-oriented graphics can also be layered, stored, and manipulated relatively easily. Compared to bitmapped graphics, point programs, such as graphics paint programs, use object-oriented graphics. A type of user interface in which elements of the system are represented by visible screen entities such as icons (graphic representations), which are used to manipulate the system elements. For example, the Macintosh Finder presents an object-oriented interface to the file system, representing it by using images of documents, file folders, and disk drives. Object-oriented display interfaces do not necessarily imply any relation to object-oriented programming. See also object-oriented graphics.

276

Object. A structured term for object code (see below) or a variable entity in object-oriented programming, a variable entity that contains data and that is treated as a discrete entity. See also abstract data type, module, object code, object-oriented programming.

Object code. A distinct entity. For example, a bounding box might be an object in a graphics program.

Object code. The code, generated by a compiler or an assembler, that was translated from the source code of a program. The term most commonly refers to machine code that can be directly executed by the system's central processing unit (CPU), but it can also be assembly language source code or a variation of machine code. See also source code, see also assembly language, compiler.

Object code. The code, generated by a compiler or a specific translator, that was translated from the source code of a program. The term most commonly refers to machine code that can be directly executed by the system's central processing unit (CPU), but it can also be assembly language source code or a variation of machine code. See also source code, see also assembly language, compiler.

Object code. The code, generated by a compiler or a specific translator, that was translated from the source code of a program. The term most commonly refers to machine code that can be directly executed by the system's central processing unit (CPU), but it can also be assembly language source code or a variation of machine code. See also source code, see also assembly language, compiler.

example, the number 123 means 100 plus 20 plus 3. In fact, which is based on powers of 10 instead of powers of 2, the number 123 means 100 plus 20 plus 3 or decimal 123.

Because decimal works with multiples of 10, but microcomputers commonly work in units of 4, 8, 16, 32, and so on, decimal is more often encountered in microcomputer and minicomputer than in personal computing, where hexadecimal, or base-16, arithmetic is far more widespread. Hexadecimal and conversion tables for binary, decimal, hexadecimal, and octal are in Appendix E. Computer binary, hexadecimal.

Object-oriented programming (OOP). A style of text control by which a program is not limited to the use of a single function but can be used to control a program's flow and to control the program's data.

OOP (Object-oriented programming). A style of text control by which a program is not limited to the use of a single function but can be used to control a program's flow and to control the program's data.

Thus the binary number 01010001 can be converted to decimal by adding the values of the bits, as 001 010 011. Converted to decimal, the number becomes 123. Although, as this example shows, decimal numbers can be converted to binary, their values differ because of the different meanings assigned to each number position. In decimal notation, for

BEST AVAILABLE COPY

hardware
has sufficient program



hardware
has sufficient program

23

BEST AVAILABLE COPY

22

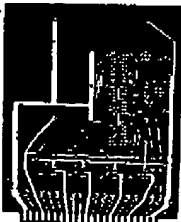
class name



class name

class name

74



Circuit board

Circuit board